AMENDMENTS TO CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Canceled)

2. (Canceled)

3. (Canceled)

4. (Canceled)

5. (Canceled)

6. (Canceled)

7. (Canceled)   Method for enabling a user to more

8. (Currently Amended) ~~The method according to claim 7~~ Method for enabling a user to edit an XML document using an XML document editor to add or delete an element into a working document and to convert said working document into an XML document file; comprising the step of enabling said XML document editor to automatically generate, in relation to two consecutive elements z_i and z_{i+1} of said working document, a list of candidate third elements z to be provided to a user; wherein said third elements z in said list are selected such that relations between elements z_i and z and between elements z and z_{i+1} comply with said DTD after said element z is inserted between elements z_i and z_{i+1}, said list enabling the user to select any of the

<u>candidate third elements z and add them to the working document without affecting DTD</u> <u>compliance of the working document</u>, wherein whether the relation between two consecutive elements complies with said DTD is determined according to the following rule:

suppose G is Glushkov Automaton of said document, z_i is a state in G, $1 \leq i \leq p-1$, [[p N]] <u>p $\in$ N</u>, $\Sigma = \{z_1, z_2, \ldots z_p\}$ is a sequence of states in G where $z_1 = s$, s is start state of G, $z_p = f$, f is final state of G;

if z_{i+1} $\in$ reachable(z_i), wherein reachable(z_i) denote the set of states in G reachable from state z_i,

then the relation between z_i and z_{i+1} is determined compliant with DTD of said document.

9. (Currently Amended) The method according to claim ~~7~~ <u>8</u>, wherein a cell C to include said candidate third elements z is generated according to the following rule and displayed as a list:

suppose (z_i, z_{i+1}) İH, H denotes the set of edges in G, G is Glushkov Automaton of regular expression E corresponding to an element of said working document;

further suppose Σ is a set to include states corresponding to all elements of G, A(E1) is the set of states in subexpression E1 to E, f-reachable(z_i) denotes the set of states in G reachable from z_i through forward edges;

if z_{i+1} $\in$ f-reachable(z_i), then let C={ z $\in \Sigma$ | z $\in$ f-reachable(z_i) and z_{i+1} $\in$ f-reachable(z)};

if z_{i+1} [[İ]] $\notin$ f-reachable(z_i), then let E1* be the smallest iteration subexpression of E that covers both z_i and z_{i+1}, C= {z $\in$ A(E1) | z $\in$ f-reachable(z_i) or z_{i+1} $\in$ f-reachable(z)}.

10. (Currently Amended) The method according to claim ~~7~~ <u>8</u>, wherein a cell C to include said candidate third ~~element~~<u>elements</u> z is generated according to the following rule and displayed as a list:

suppose (z_i, z_{i+1}) $\in$ H, H denotes the set of edges in G, G is Glushkov Automaton of regular expression E corresponding to an element of said working document;

further suppose Σ is a set to include states corresponding to all elements of G, A(E1) is the set of states in subexpression E1 to E, f-reachable(z_i) denotes the set of states in G reachable from z_i through forward edges;

if (z_i, z_{i+1}) is a forward edge, let $C=\{\ z \in \Sigma \mid z \in$ f-reachable(z_i) and $z_{i+1} \in$ f-reachable$(z)\}$ and:

if $z_i \in$ last$(E1*)$ for some iteration subexpression $E1*$ of E and E1 is the largest one, then let $C1= \{z \in A(E1) \mid z_{i+1} \in$ f-reachable$(z)\}$, $C= C \cup C1$;

if $z_{i+1} \in$ first$(E2*)$ for some iteration subexpression $E2*$ of E and E2 is the largest one, then let $C2= \{z \in A(E2) \mid z \in$ f-reachable$(z_i)\}$ and $C= C \cup C2$;

if (z_i, z_{i+1}) is a backward edge, then let $C= A(E3)$, wherein $E3*$ is the largest iteration subexpression of E satisfying $z_i \in$ last$(E3)$ and $z_{i+1} \in$ first$(E3)$.

11. (Currently Amended) The method according to claim ~~7~~ 8, further comprising automatically generating a required element between element pair z_i and z and element pair z and z_{i+1} after said third element z is inserted between element pair z_i and z_{i+1}, such that said working document is effective; wherein said requirement comprises articulation points between elements z_i and z (and z and z_{i+1}) in Glushkov Automaton G; i.e., states through which all paths between z_i and z (and z and z_{i+1}) shall pass.

12. (Currently Amended) The method according to claim ~~7~~ 8, further comprising automatically generating an element slot allowing user to add elements into said document, if no required element between element pair z_i and z and element pair z and z_{i+1} is found after said third element z is inserted between element pair z_i and z_{i+1} and if $(z_i, z) \ddot{I} H$ $((z, z_{i+1}) \ddot{I} H)$, wherein H denotes set of edges in G; and wherein said requirement comprises articulation points between elements z_i and z (and z and z_{i+1}) in Glushkov Automaton G; i.e., states through which all paths between z_i and z (and z and z_{i+1}) shall pass.

13. (Canceled)

14. (Canceled)

15. (Canceled)

16. (Canceled)

17. (Canceled)

18. (Canceled)